**Rapport de Projet Bases de données semi-structurées : Technologie XML (XSD, XSL/XPath, XQuerry)**

**Présenté par : Mouad CHOUKI DATA INE1**

**Rappel de l’objet du projet :**

L'Objectif du projet est Concevoir et à mettre en œuvre un système de gestion de bibliothèque numérique entièrement fonctionnel en utilisant des technologies XML telles que XSD, XSLT, XPath et XQuery/XUF.

- Représenter les différentes entités de la bibliothèque comme les documents, les auteurs, les catégories, les emprunteurs, etc...

- Proposer des fonctionnalités de gestion comme recherche, impression, états de sortie, etc....

**1. Modélisation des Données**

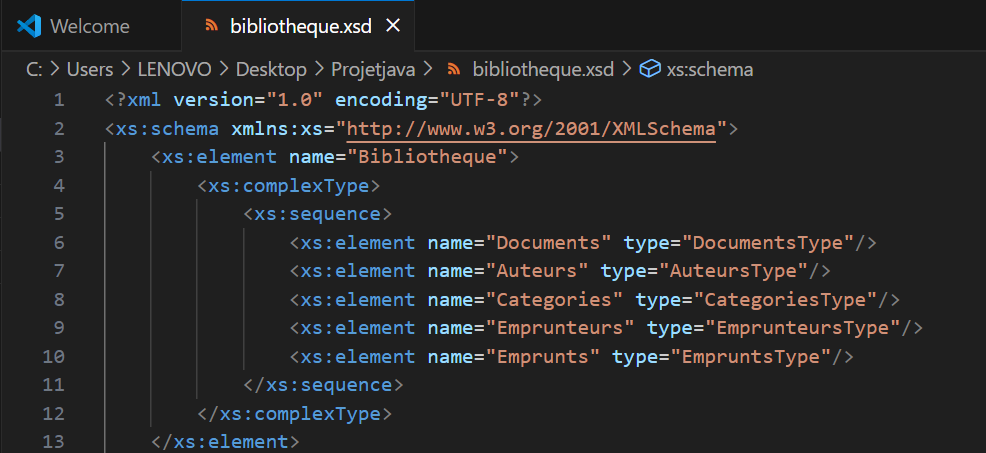
**1.1. Définir les Entités**

Les entités principales à représenter dans le système sont :

* **Document** : titre, auteur, catégorie, date de publication, ISBN (International Standard Book Number), Nombre d’exemplaire, Nombre d’emprunts.
* **Auteur** : nom, date de naissance, nationalité, etc.
* **Catégorie** : nom, description, etc.
* **Emprunteur** : nom, adresse, numéro de téléphone, email, etc.
* **Emprunt** : Id d’emprunt, emprunteur, document, date d'emprunt, date de retour, etc.

**1.2. Créer le Schéma XML (XSD)**

Le schéma XML (XSD) va définir la structure des documents XML utilisés dans le système dans un fichier bibliotheque.xsd, l’extension .xsd nous permettra de le lier prochainement à un fichier avec une extension .xml.

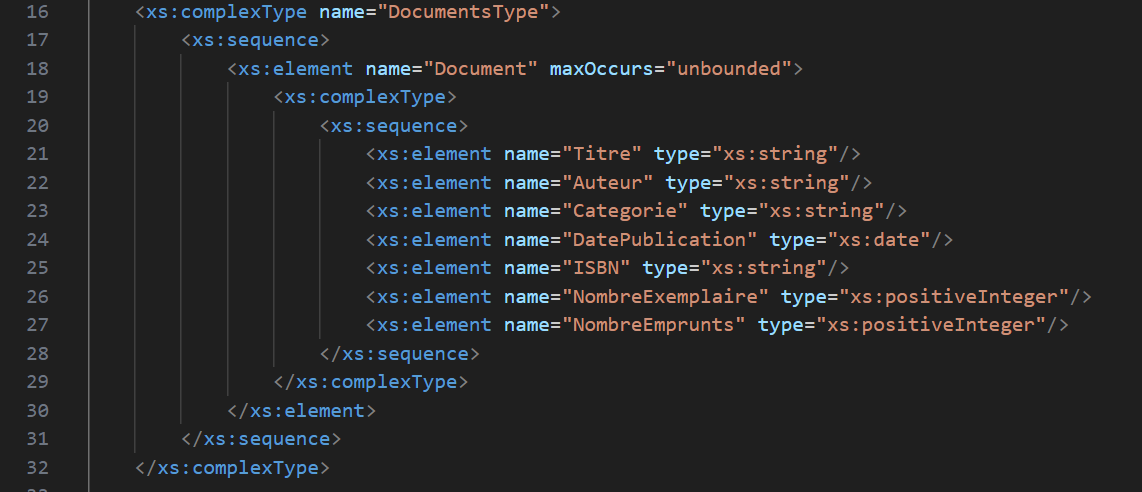


Ce code définit les éléments principaux que nous retrouverons dans notre bibliothèque numérique, il faudra à présent définir les différents ComplexType DocumentsType, AuteursType, CategoriesType, EmprunteursType, EmpruntsType. Voici le code XSD pour les différents types.

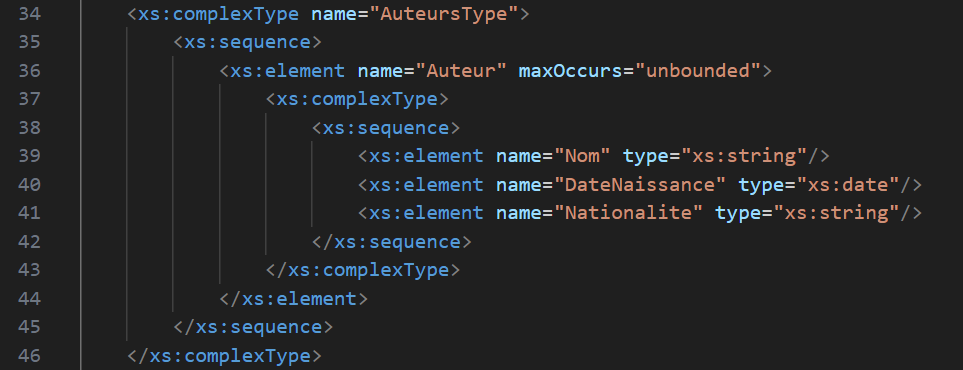


1.2.1. Schéma XSD du Complexe Type DocumentsType :

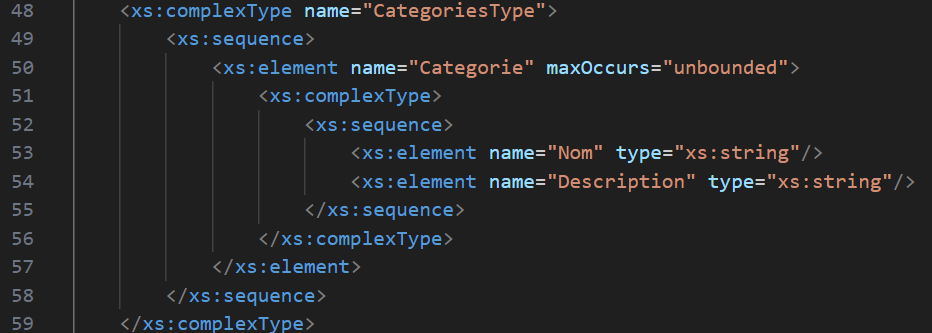
NB : l’élément ISBN est un numéro d’identification d’article selon des normes bien définies



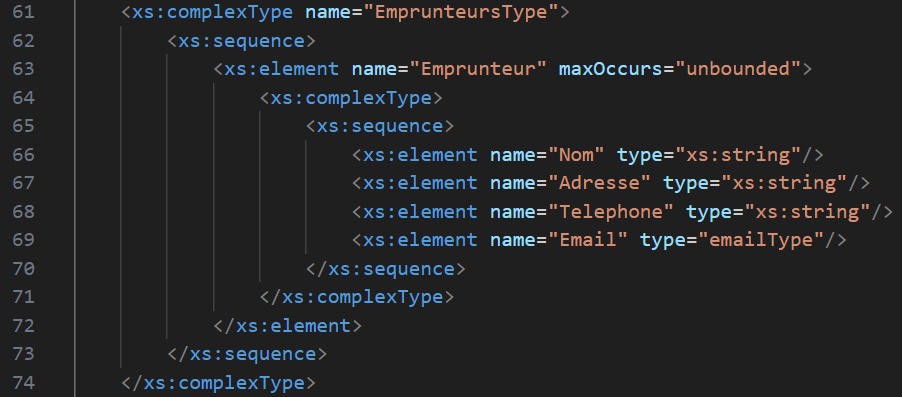
1.2.2. Schéma XSD du Complexe Type AuteursType :



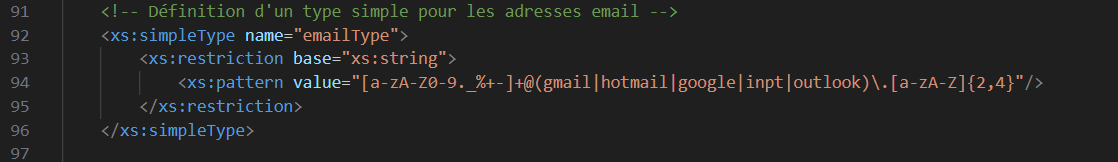
1.2.3. Schéma XSD du Complexe Type CategoriesType :



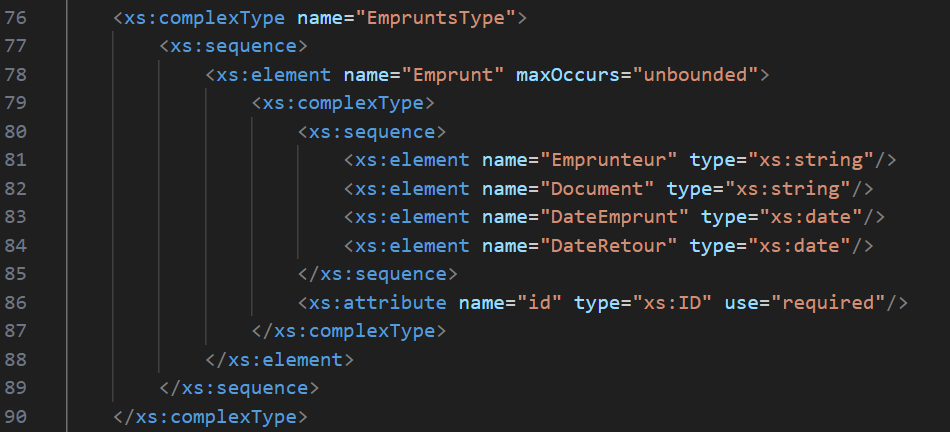
1.2.4. Schéma XSD du Complexe Type EmprunteursType :



J’ai aussi rajouté un Simple Type email qui permettra que les adresses mails des emprunteurs soient des adresses mails valides



1.2.5. Schéma XSD du Complexe Type EmpruntsType :

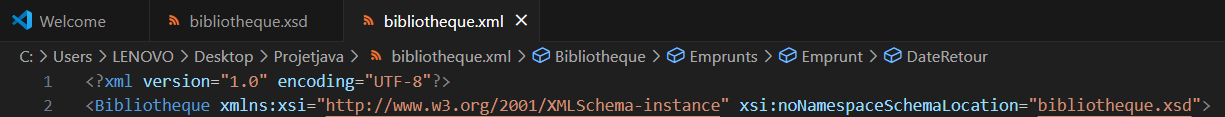


Maintenant que les schémas XSD sont bien définis, nous allons créer un document XML pour instancier les différents éléments documents, auteurs, catégories, emprunteurs et emprunts.

### 2. Création de Documents XML

Exemple d'un document XML pour la bibliothèque :

Commençons d’abord par lier le fichier bibliotheque.xml au fichier bibliotheque.xsd afin que nos instanciations respectent le schéma xsd prédéfini.

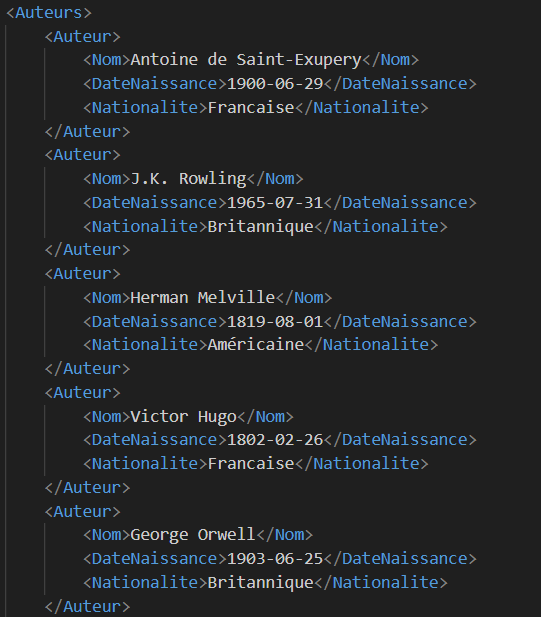


Nous pouvons à présent commencer à alimenter notre bibliothèque numérique :

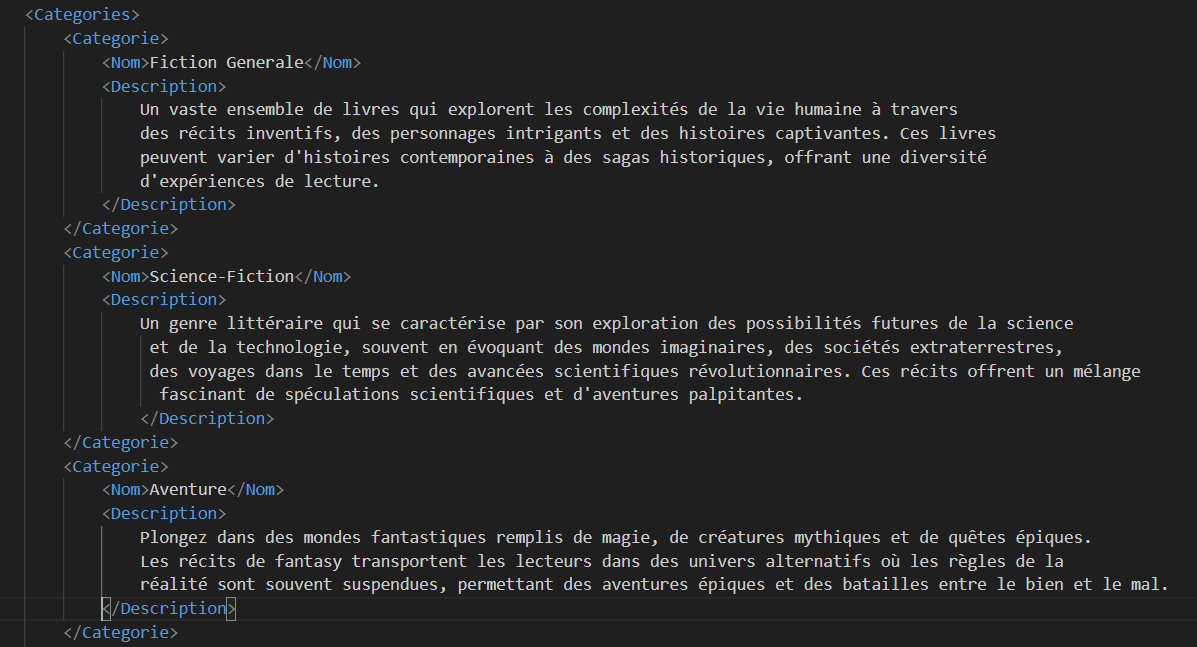
**2.1. Exemples de Documents**



**2.2. Exemples d’auteurs**

****

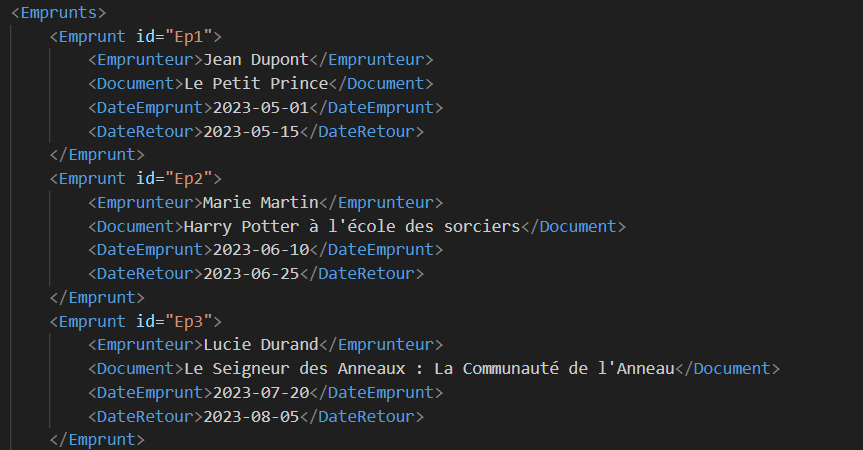
**2.3. Exemple de catégories**



**2.4. Exemple d’emprunteurs(clients)**



**2.5. Exemples d’emprunts**



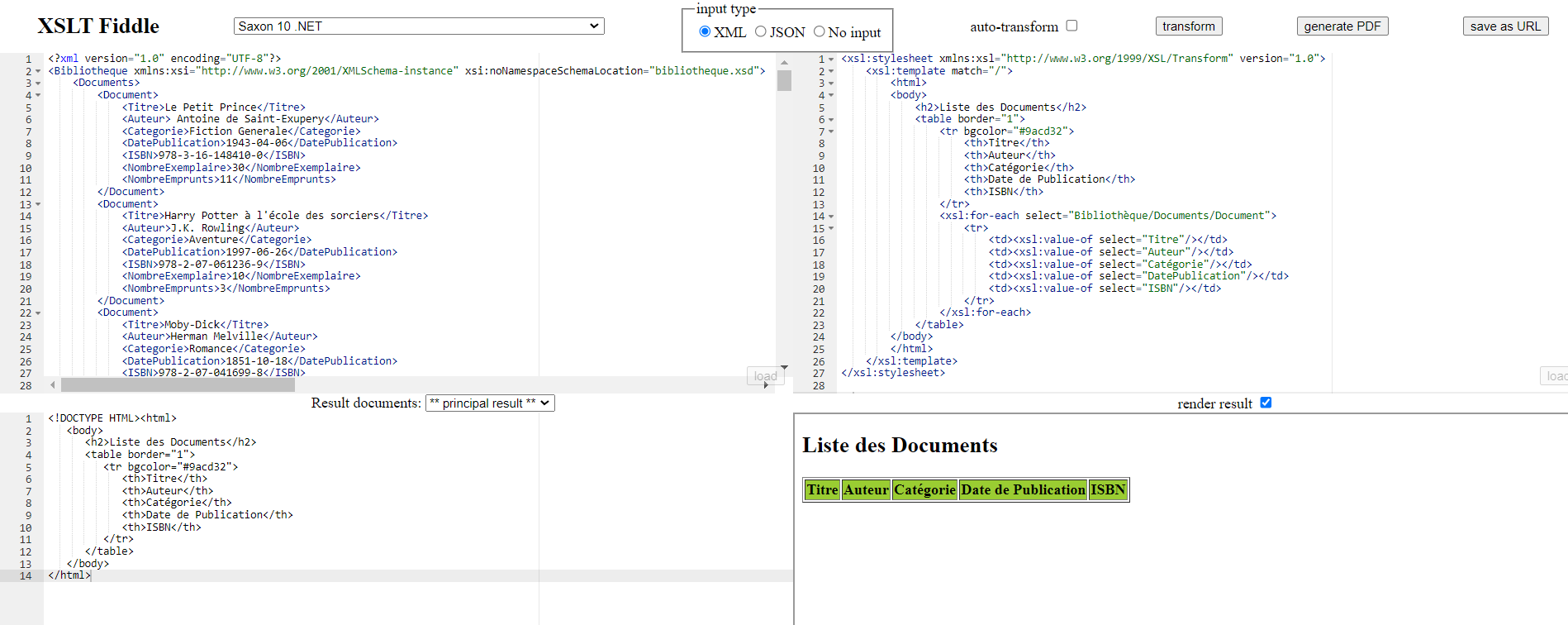
A noter que pour chaque élément il y’a eu beaucoup plus d’instanciations, mais faute d’espace et afin que le rapport ne soit pas trop long à lire j’ai décidé de ne mettre que quelques exemples pour chaque élément, je joindrai également mes codes avec le rapport afin que le lecteur puisse les examiner s’il le souhaite.

Passons à présent à la feuille XSLT qui nous servira de transformateur à notre document bibliotheque.xml afin que l’on puisse visualiser comme l’on veut notre base de données à travers un fichier html.

### 3. Créer la Transformation XSLT

**3.1. Test préliminaire**



Voilà le code xsl préliminaire que j’ai utilisé pour transformer mon fichier bibliotheque.xml en résultat html grâce à l’outil XSLT Fiddle. 

Le code du document bibliotheque.xml a été mis en haut à gauche, le code xsl en haut à droite et il suffisait ensuite de cliquer sur Transform et le résultat est généré en bas à droite de l’écran avec un le code html qui lui correspond en bas à gauche.

**3.2. Ajout de format et de restrictions**

Maintenant que je sais que ma structure marche et que mon code xsl est bien défini, je vais essayer de l’améliorer afin qu’il m’affiche mes documents rangés selon leurs catégorie avec un code couleur pour chaque catégorie, puis je voudrais aussi qu’il m’affiche la liste des auteurs dont les articles sont présents dans ma bibliothèque avec la particularité que les auteurs français soient en bleu, et enfin je vais lui demander de l’afficher tous les emprunts enregistrés dans ma bibliothèque dans une table à part

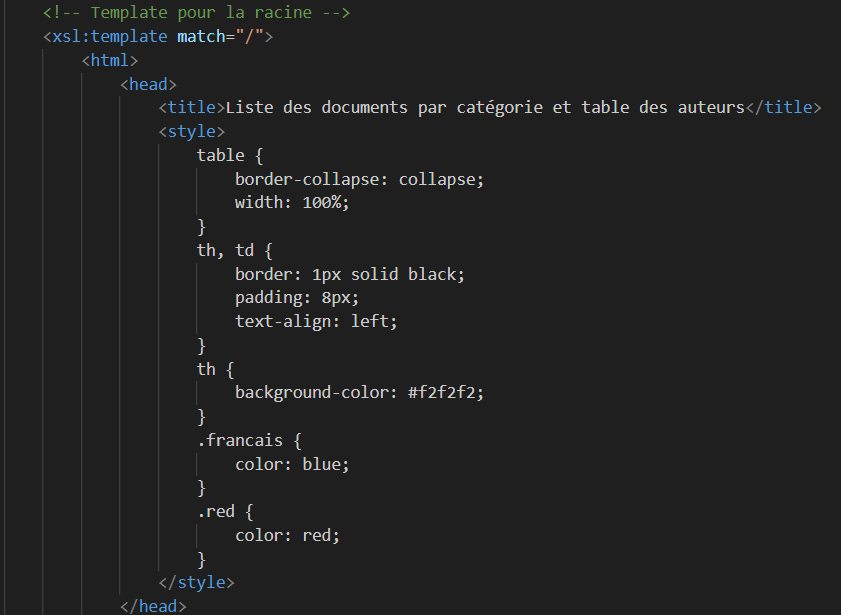
Voici le code qui m’a permis de réaliser tout cela :

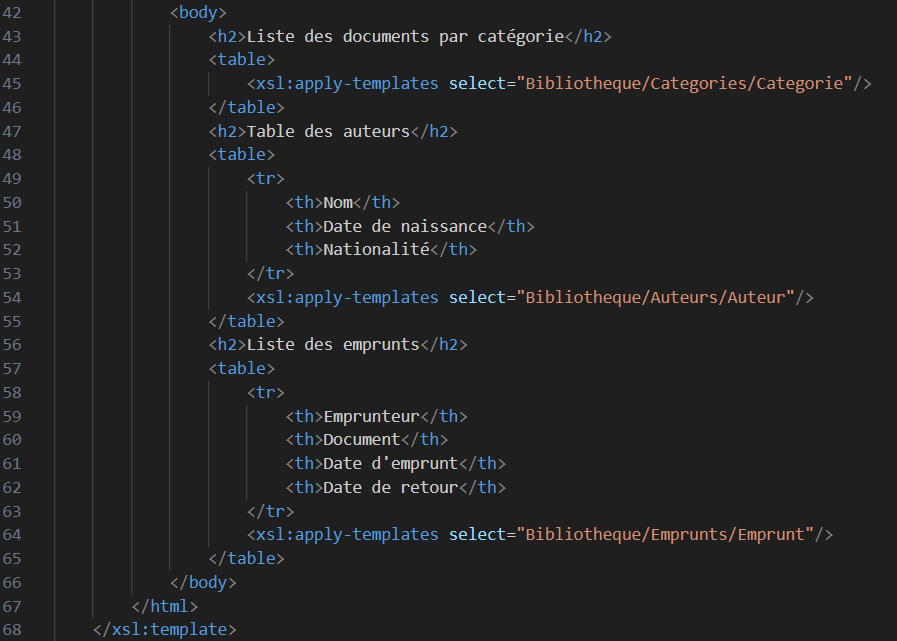
3.2.1. Codage de l’affichage des documents :

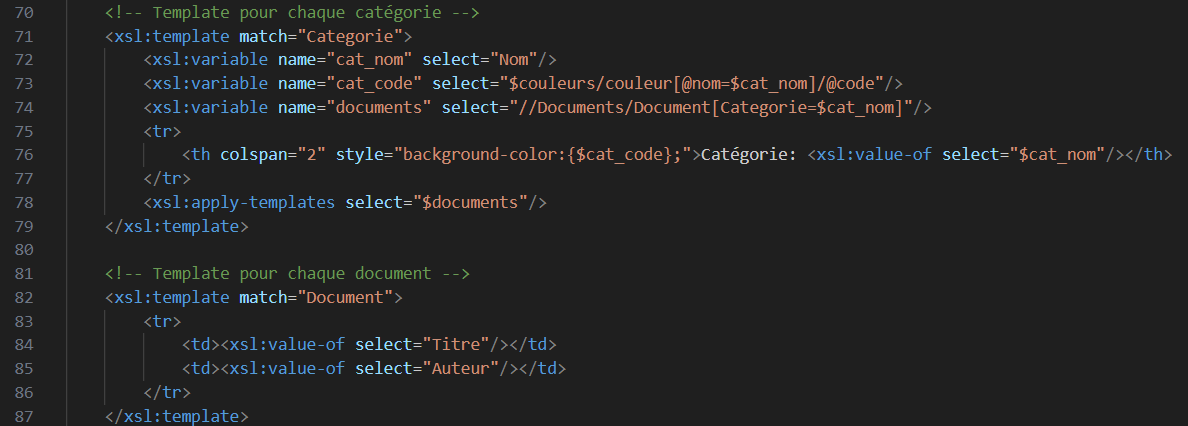
Définition du code couleur :

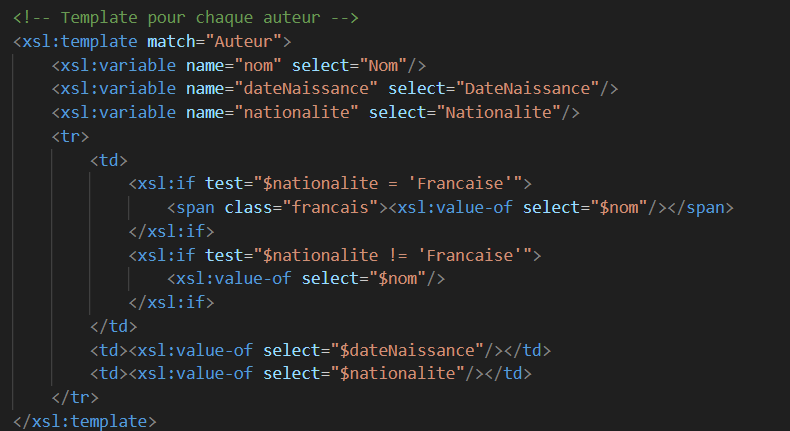


Codage de l’affichage :

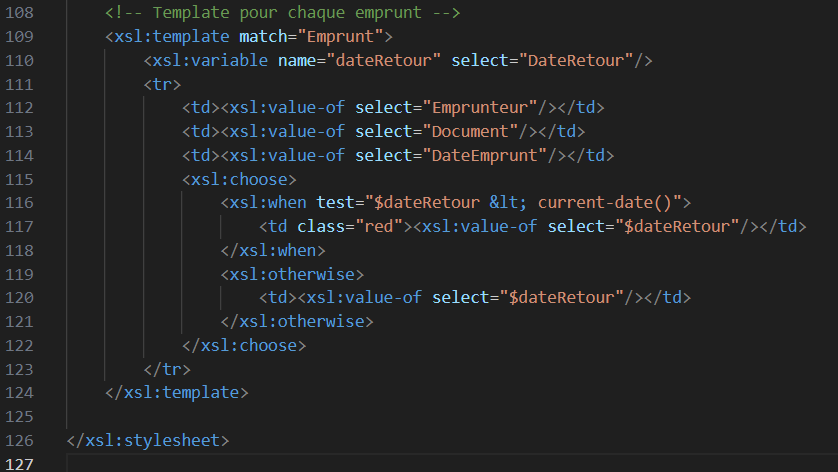






3.2.2. Codage de l’affichage des auteurs

3.2.3. Codage de l’affichage des emprunts



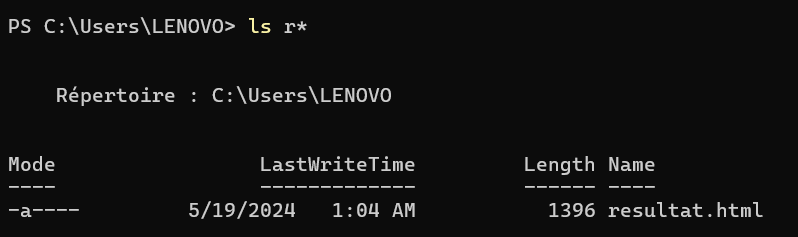
**4. Visualiser le résultat HTML**

Maintenant que tous nos codes sont prés il va falloir visualiser notre résultat. Pour se faire j’ai installé le processeur Saxon, qui contient un fichier jartype saxone9he.jar, qui permet la transformation effective du fichier bibliotheque.xml à travers le fichier bibliotheque.xsl et qui met le résultat dans un fichier resultat.html dont l’exécution aboutit directement sur l’affichage attendu dans un navigateur Web.

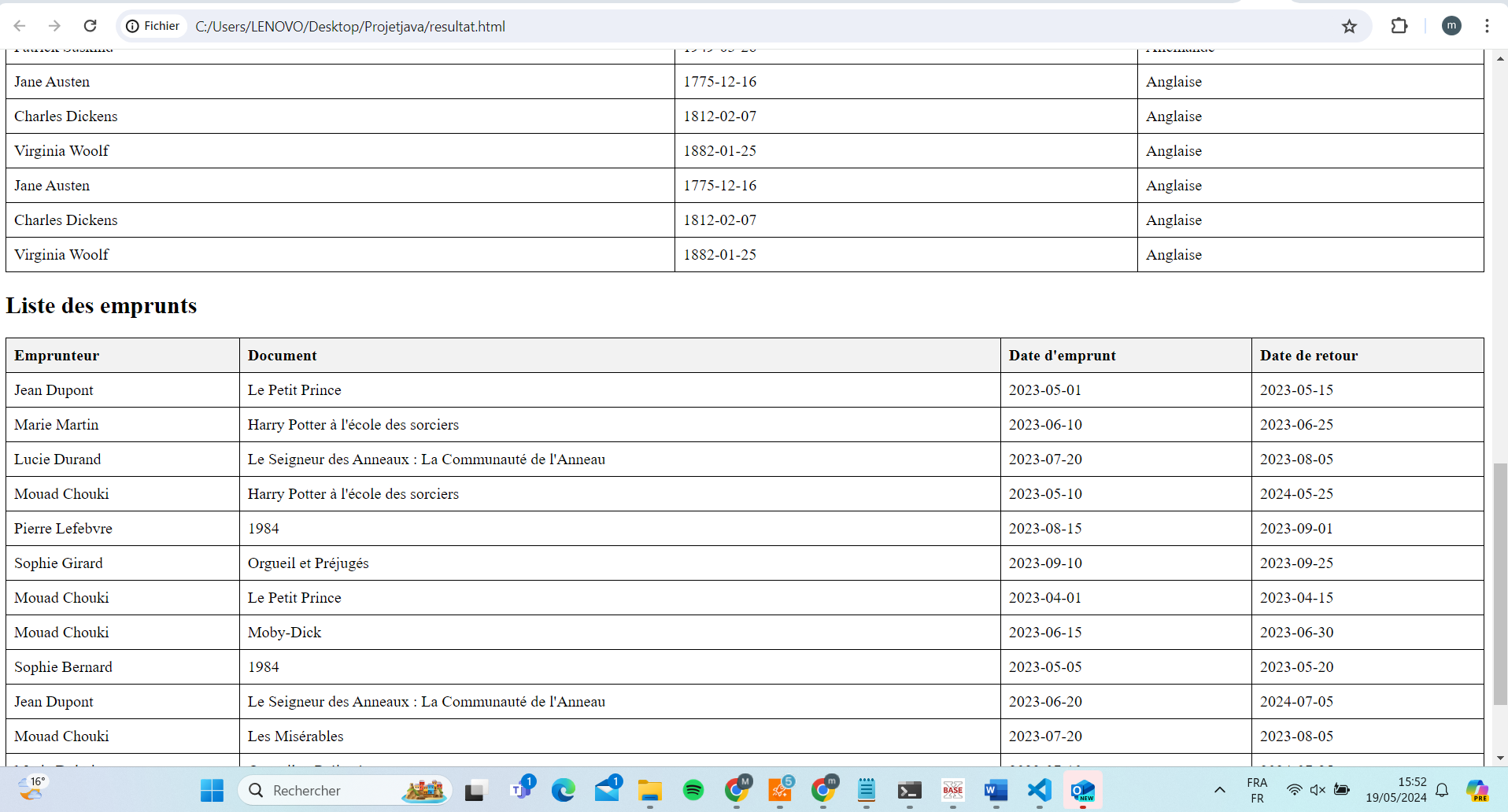
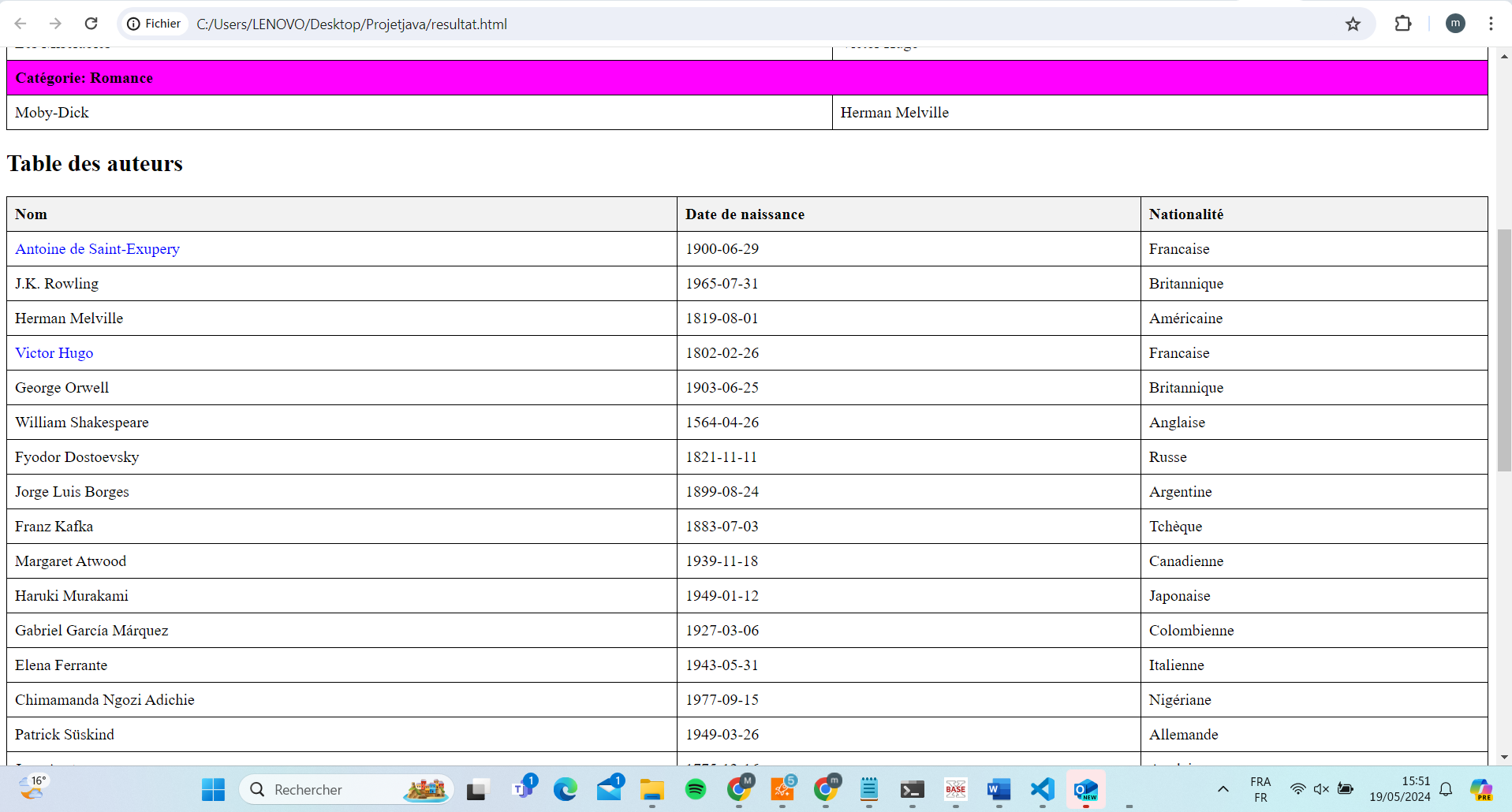
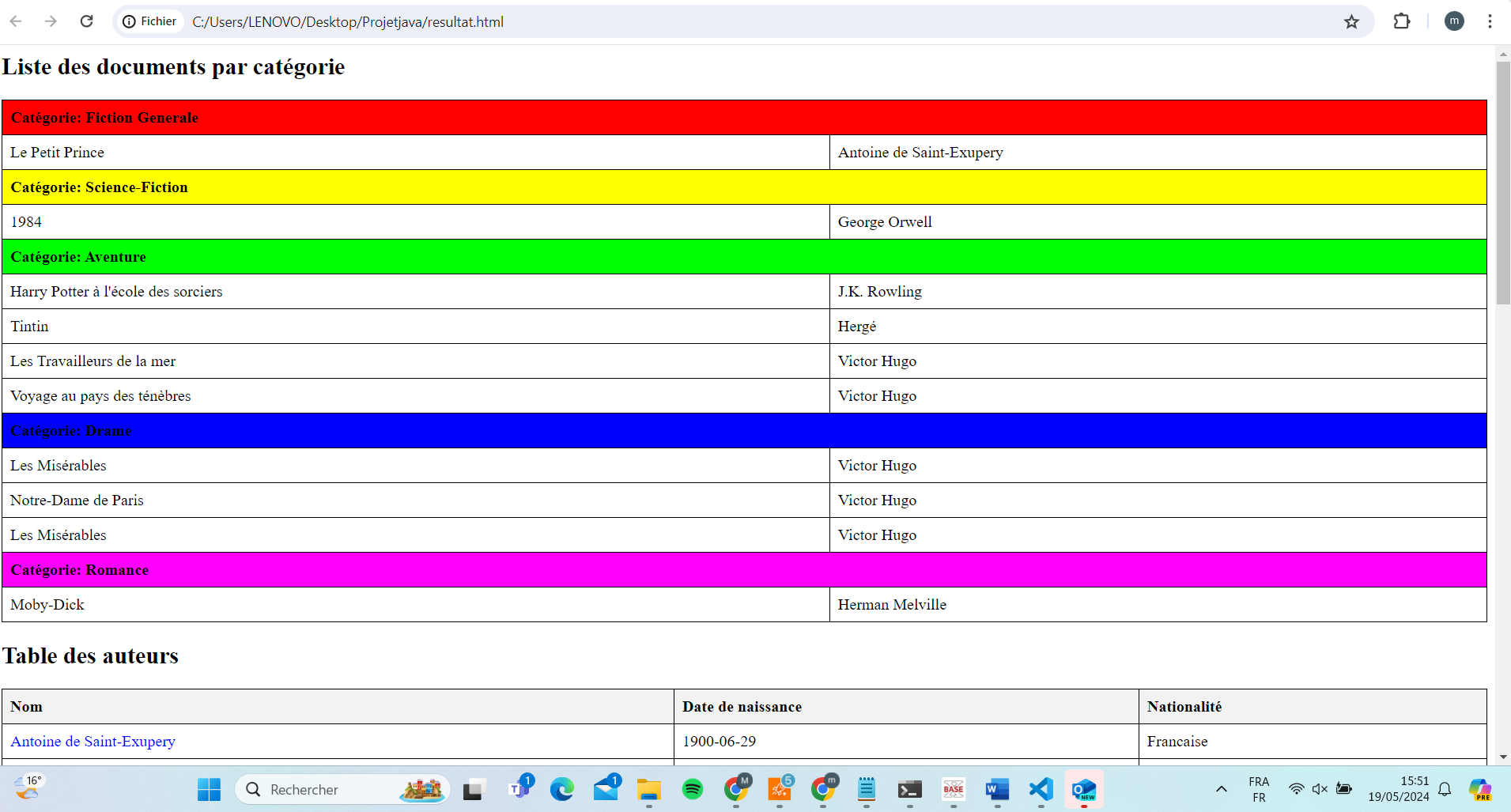
Une fois l’installation faite, il suffit d’exécuter sur notre terminal la commande suivante :



Nous retrouvons ensuite le fichier qui a bien été créé avec succès

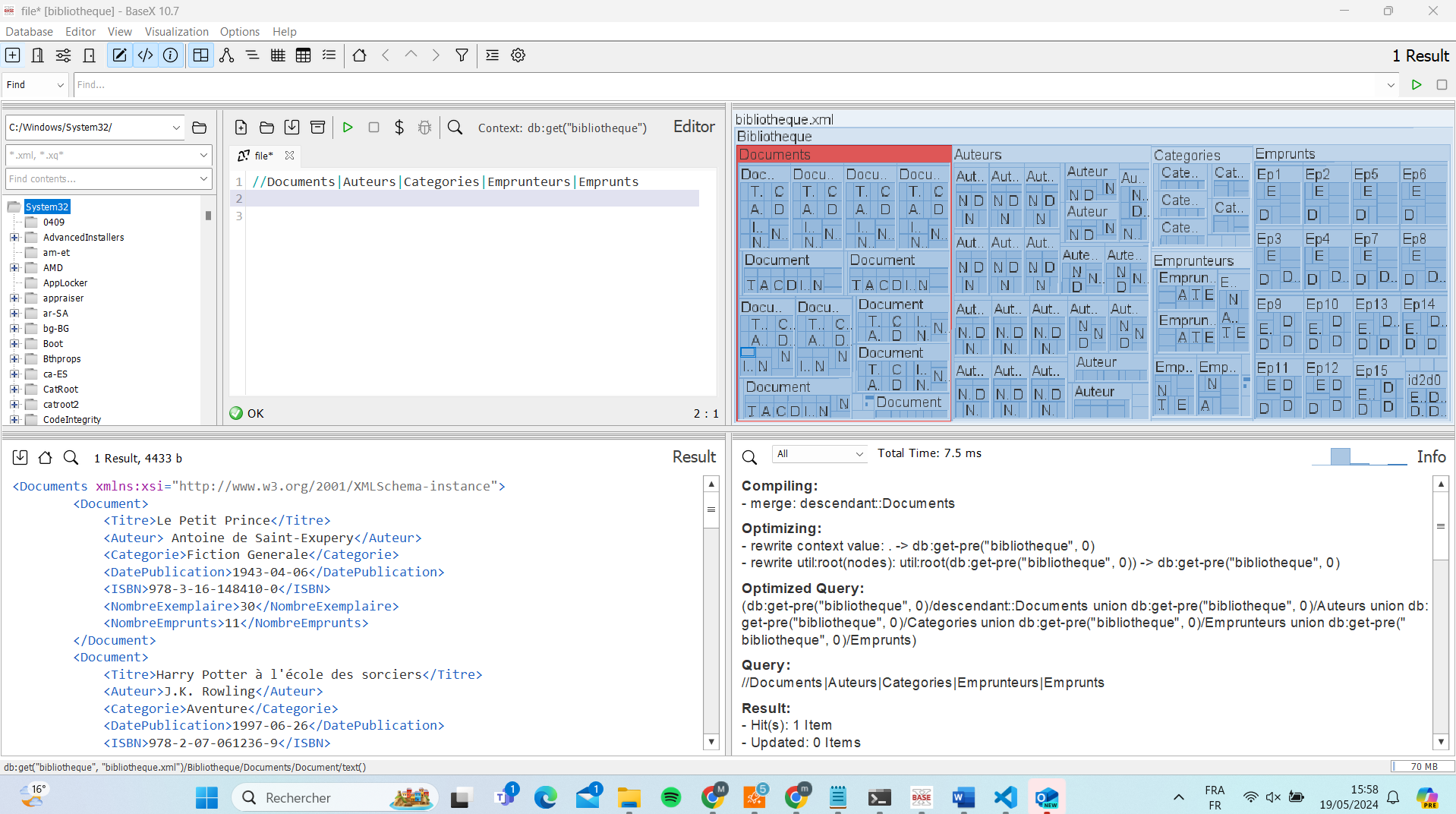


Il ne reste plus qu’à l’ouvrir afin de vérifier si notre configuration a été respecté



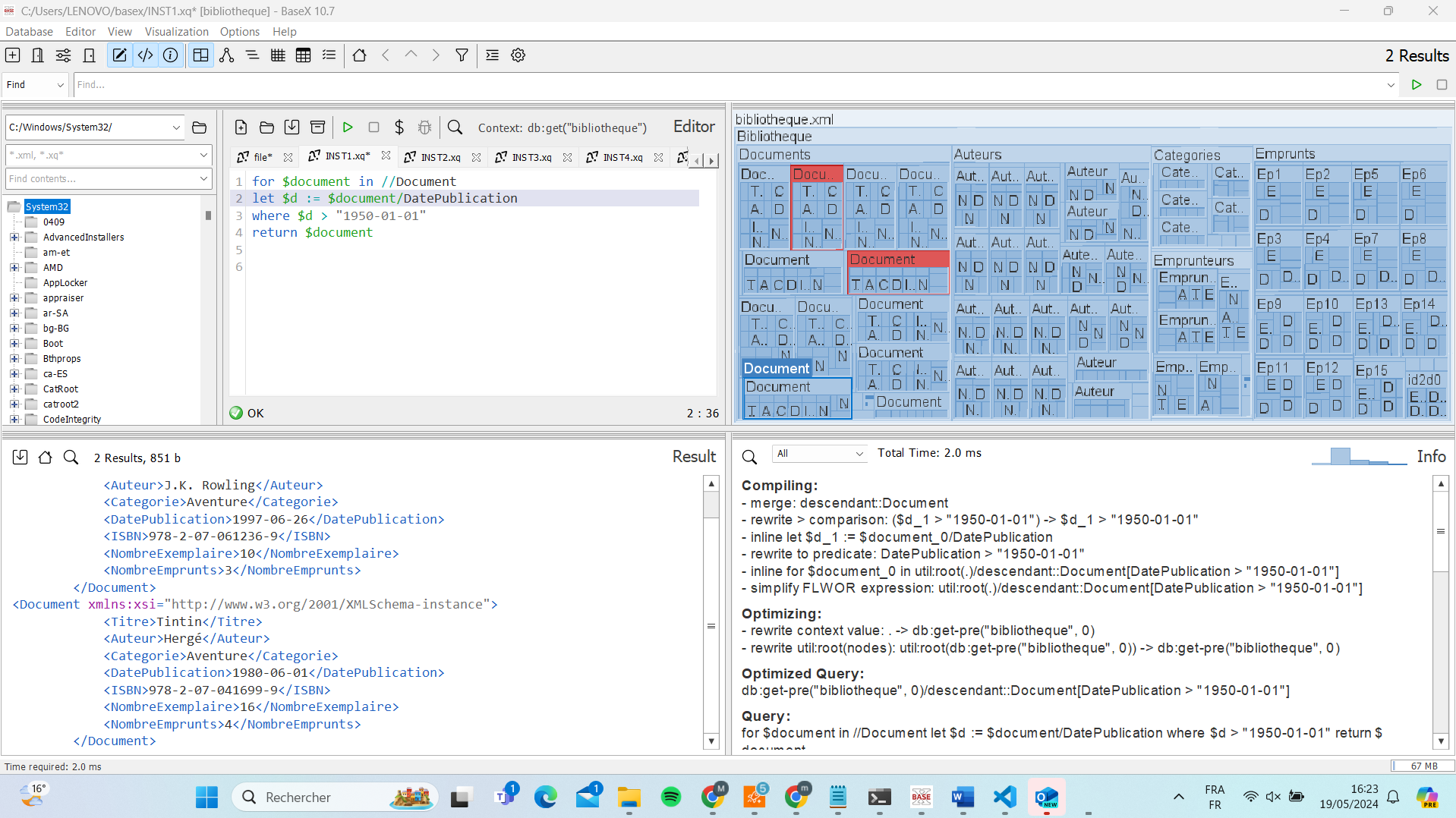
Effectivement notre configuration a bien été respecté.

5**.Exemple de gestion de la bibliothèque numérique avec XQuerry sur BaseX :**

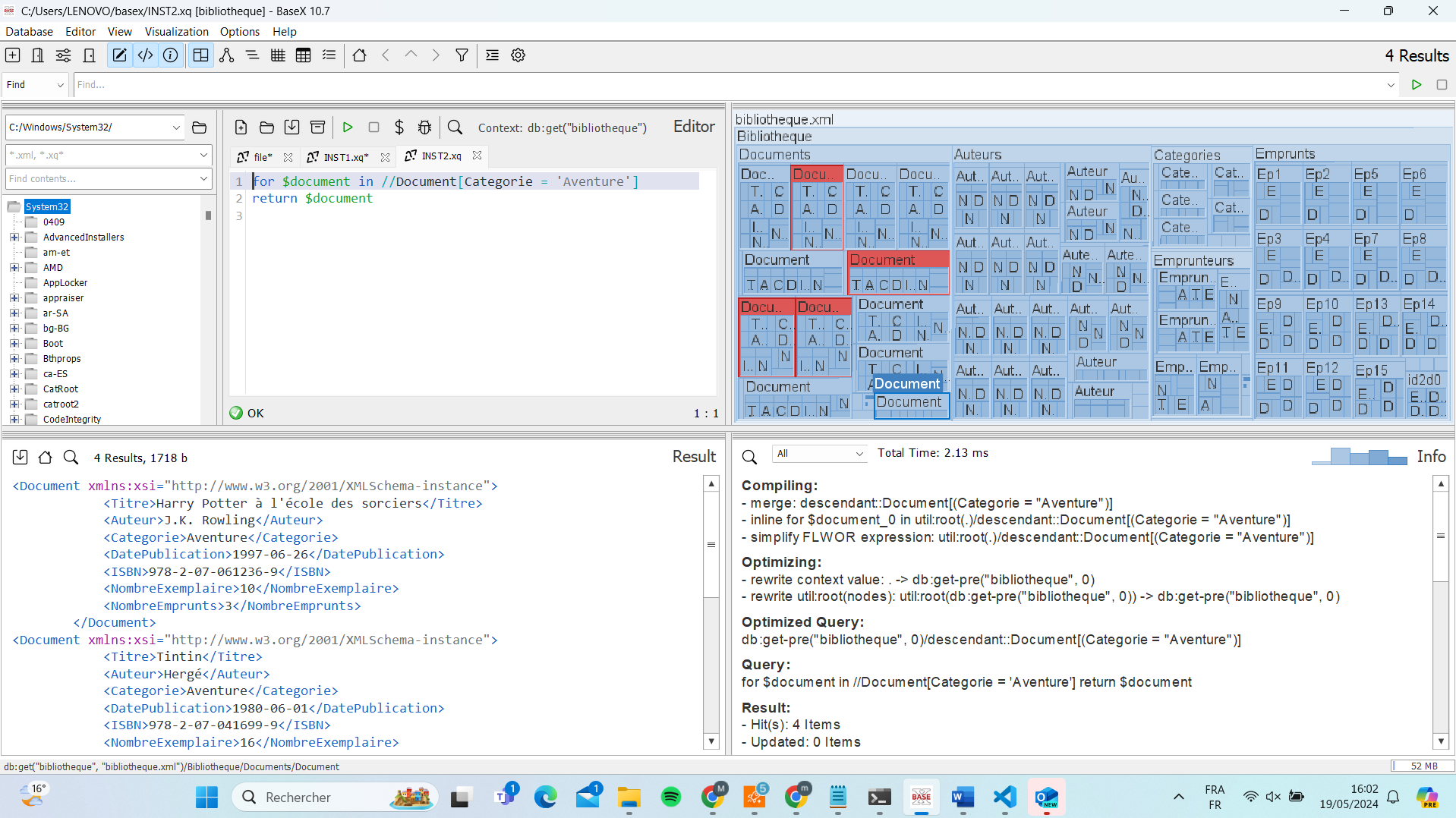
****

Je propose ici 5 instructions de gestion :

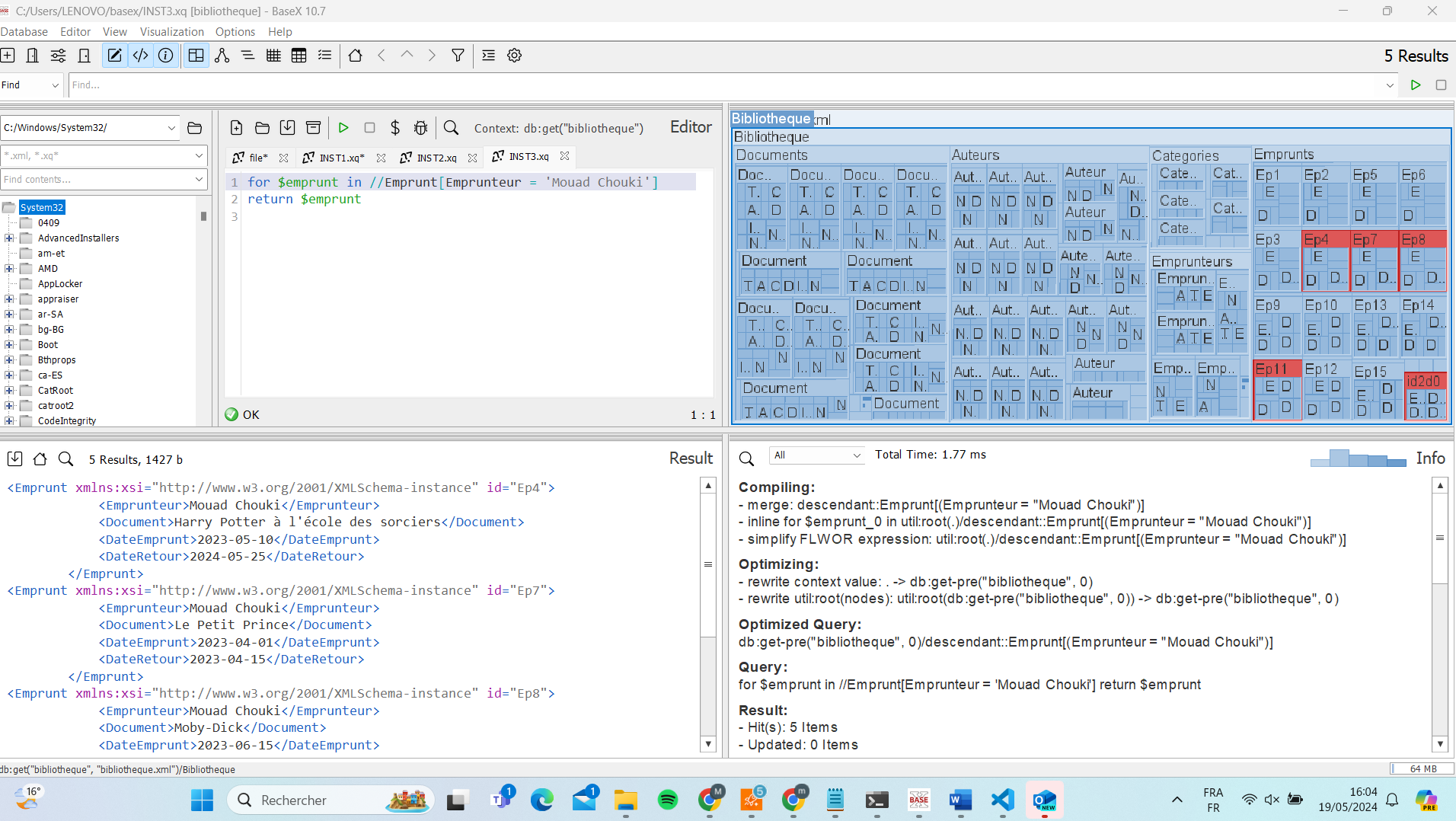
* 1. **Lister tous les documents parus après 1950 :**



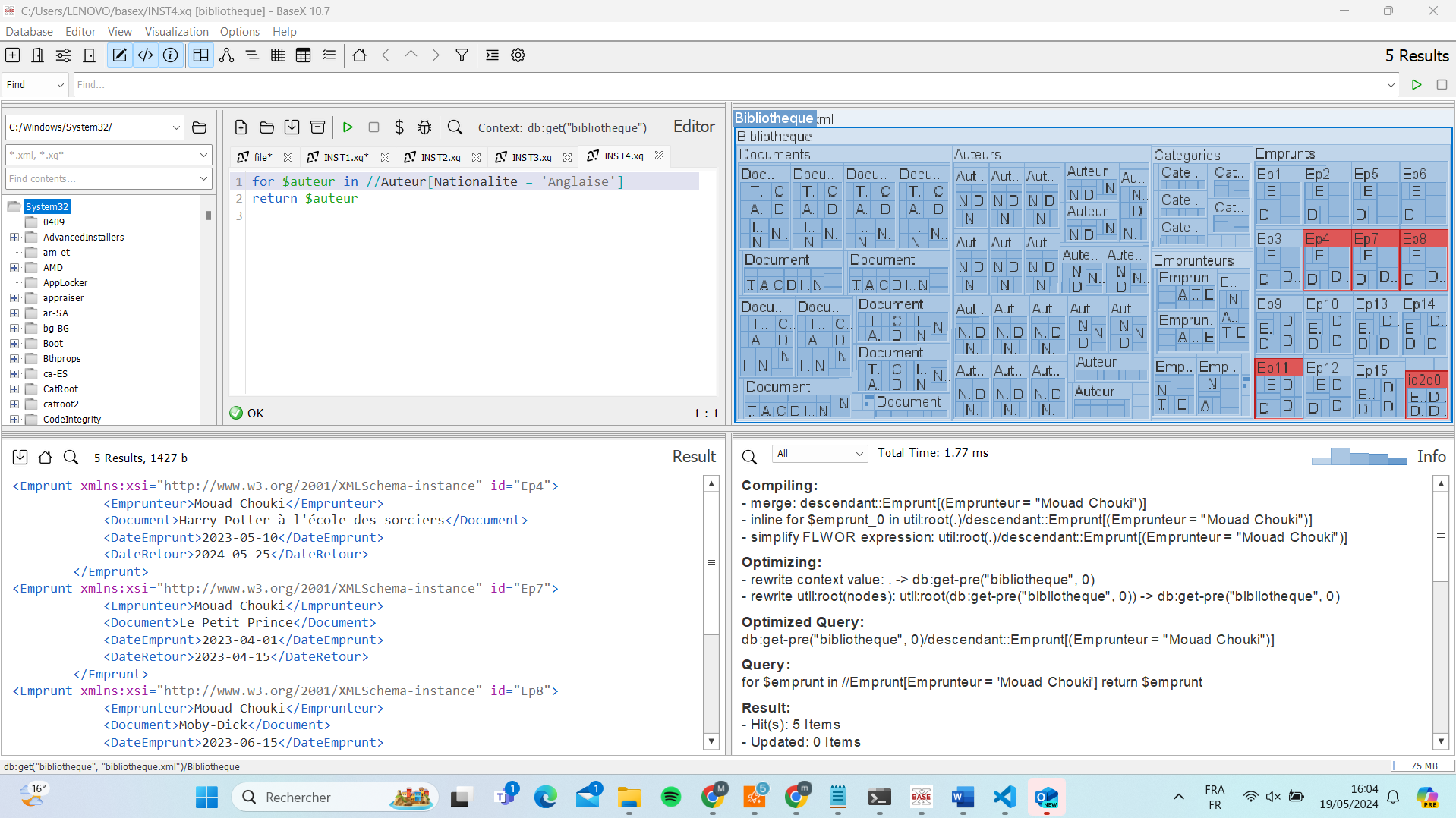
* 1. **Boucler sur la catégorie Aventure et afficher tous les documents qui sont de cette catégorie :**



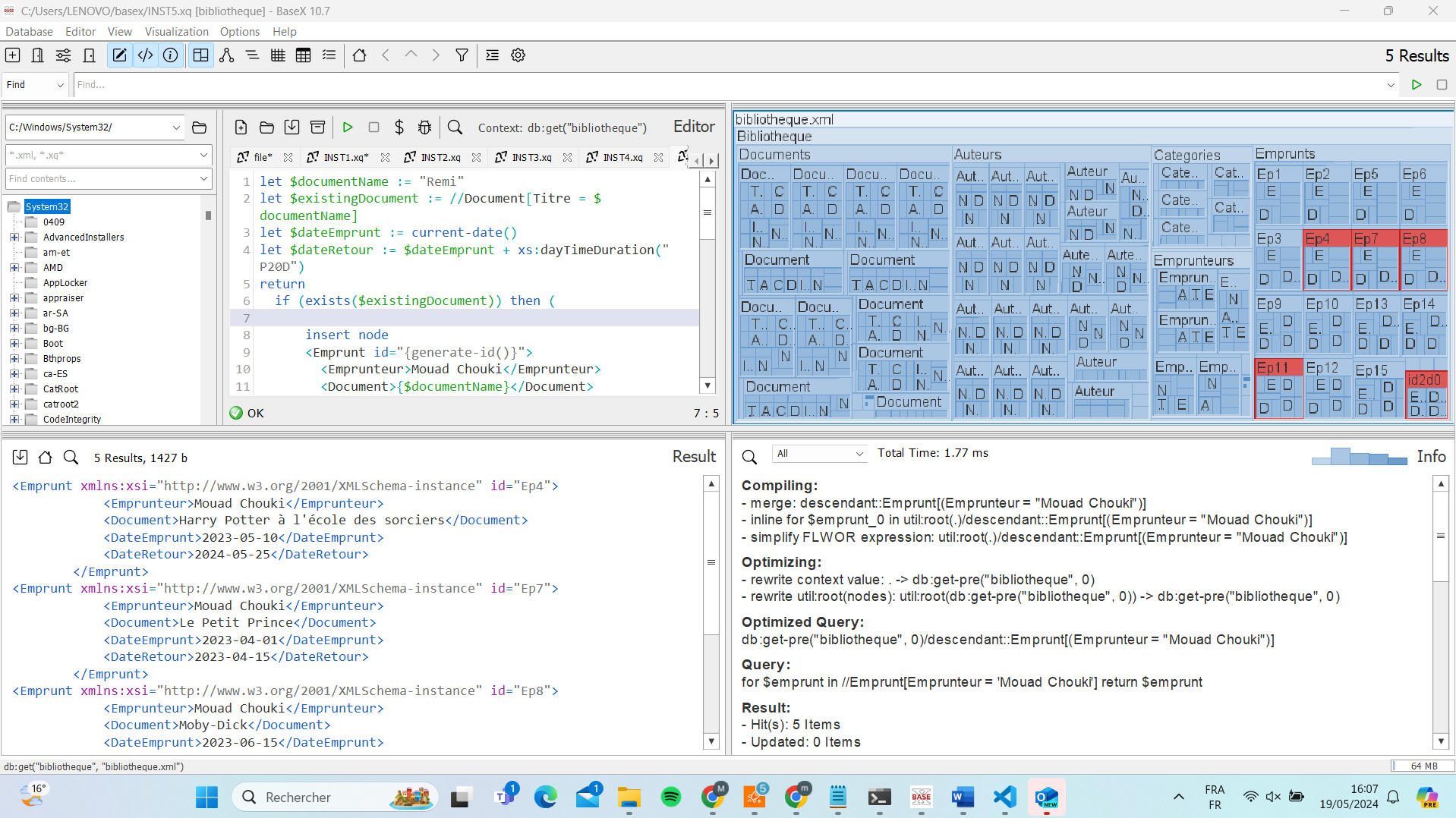
* 1. **Afficher tous les emprunts effectués par Mouad CHOUKI :**



* 1. **Afficher tous les auteurs de nationalité anglaise :**



* 1. **Ajouter un emprunt du document Rémi par Mouad Chouki et incrémenter le nombre d'emprunts si le document existe, sinon ajouter le document avec un nombre d'emprunts initialisé à 1 :**



Je joindrais avec mon rapport les fichiers correspondant à chaque requête également.

**6.Quelques remarques :**

Nous avons ici essayé de créer une bibliothèque numérique entièrement grâce à la technologie XML enseigné en cours de base de données semi-structurées, mais nous remarquerons que le travail reste perfectible notamment en faisant en sorte que les codes supportent les accents comme le ‘é’ le ‘è’ car en effet dès que j’en rajoutais mes codes buggais. L’affichage HTML reste objet d’amélioration constante en affichant les emprunts dont la date de retour est dépassée en rouge par exemple, mais l’objectif n’était pas là, nous avons juste montré comment manipuler xsl afin de pouvoir contrôler notre affichage. J’ai aussi essayé d’utiliser un maximum de notions vu en cours comme l’énumération des attributs possibles, ou la mise en forme d’une structure spécifique de chaine de caractère comme je l’ai fait pour vérifier la validité des adresses mails, j’ai aussi utilisé la convention FLWOR ( For Let While Order Return) pour mieux contrôler mes requêtes XQuerry, en bref l’objectif était de passer en revue toutes les capacités acquises et de les mettre en application, je vous remercie de m’avoir lu en espérant que ce projet saura gagner votre satisfaction.

**Fait par Mouad CHOUKI**

**DATA INE1**